

Flight Testing of Guidance, Navigation and Control Systems on the Mighty Eagle Robotic Lander Testbed

Mike Hannan¹, Doug Rickman², Greg Chavers³, Jason Adam, Chris Becker, Joshua Eliser⁴, Dan Gunter, Logan Kennedy, Patrick O'Leary⁵

NASA/Marshall Space Flight Center, Huntsville, AL 35812

During 2011 a series of progressively more challenging flight tests of the Mighty Eagle autonomous terrestrial lander testbed were conducted primarily to validate the GNC system for a proposed lunar lander. With the successful completion of this GNC validation objective the opportunity existed to utilize the Mighty Eagle as a flying testbed for a variety of technologies. In 2012 an Autonomous Rendezvous and Capture (AR&C) algorithm was implemented in flight software and demonstrated in a series of flight tests. In 2012 a hazard avoidance system was developed and flight tested on the Mighty Eagle. Additionally, GNC algorithms from Moon Express and a MEMs IMU were tested in 2012. All of the testing described herein was above and beyond the original charter for the Mighty Eagle. In addition to being an excellent testbed for a wide variety of systems the Mighty Eagle also provided a great learning opportunity for many engineers and technicians to work a flight program.

1. Introduction

Since 2010, the Marshall Space Flight Center (MSFC) has been testing a monopropellant robotic lander known as the Mighty Eagle, previous named the Warm Gas Test Article (McGee et al. 2013). This system, collaboratively built with the Johns Hopkins University Applied Physics Laboratory (APL), was developed to test technologies used in terminal descent on a variety of airless bodies – the final 30-60 meters. This supports both the Exploration Missions Systems and the Science Mission Directorates of NASA. Dynamically, the Mighty Eagle was designed to be similar to a lander planned to deliver scientific instrumentation for the International Lunar Network (ILN) project.

The technologies tested on-board the Mighty Eagle span the gamut of those needed for successful landing of a small robotic lander:

- Advanced Propulsion - Pulsed Thrusters,
- Guidance, Navigation & Control,
- Advanced Thermal Materials - High-Strength, Light-Weight Structures & Energy Absorbing Legs,
- Utilization of Advanced Electrical Power Technologies.

Flight test campaigns were performed in 2011, 2012 & 2103. In 2011 the GNC system was validated in a series of envelope expanding flight tests. The APL optical velocimetry system was tested on the Mighty Eagle at the end of 2011. In 2012 the optical camera used for velocimetry was re-tasked to provide images for real-time testing of autonomous rendezvous and capture (AR&C) algorithms. In 2013 an optical hazard avoidance system was flight tested over a high fidelity lunar terrain field. Also in 2013 Moon Express validated parts of their GNC system on two Mighty Eagle flights. A low cost MEMS IMU was installed for the final two flights of 2013 and its performance evaluated.

¹ mike.r.hannan@nasa.gov, NASA/MSFC

² Scientist, Earth Science Office, NASA/MSFC

³ RLLDP Program Manager, NASA/MSFC

⁴ Leidos Inc., Huntsville, AL 35805

⁵ Leidos Inc., Huntsville, AL 35805

2. Vehicle

2.1. Hardware

The three footpads of the lander define a triangle with sides 2.13 m long (Fig. 1). It has a dry mass of approximately 207 kg, and flies with up to 116 kg of fuel and 7 kg of pressurant. Balance masses are part of the vehicle, which allows compensation for changes in payload mass and distribution. Maximum nominal flight time is 47 seconds. All lander materials were chosen to meet compatibility standards with the hydrogen peroxide propellant.



Figure 1. Mighty Eagle on the steel launch plates during pre-flight warm up.

The propulsion system is a monopropellant, pressure-regulated system. Pressurized, high-purity nitrogen forces 90% hydrogen peroxide into the reaction chamber. Disassociation of the H_2O_2 is catalyzed by silver screens. There are twelve 53.4 N (12 lbs), pulsed, attitude control system (ACS) thrusters, three 330 N (74 lbs), pulsed, descent thrusters, and one throttleable Earth Gravity Cancelling (EGC) thruster with a maximum thrust of approximately 2710 N (609 lbs). The ACS thrusters are grouped into six coupled pairs to allow torque to be applied independently to each of the three rotation axes of the vehicle. The descent thrusters control the ascent/descent of the vehicle. The EGC, meant to “null out” the difference between the “desired” gravity and earth gravity, throttles down as fuel is used and the vehicle becomes lighter. All tests conducted so far have simulated lunar gravity ($1/6^{\text{th}}$ of earth gravity), so the EGC attempted to produce thrust equal to $5/6^{\text{ths}}$ of the vehicle’s weight.

The base GNC sensor suite (Fig. 2) consists of an inertial measurement unit (IMU) (Northrop Grumman LN200), a radar altimeter (Roke Miniature Radar Altimeter), and three ground contact switches. A GPS unit (Novatel ProPak-V3-RT2j) is installed, but is only used by the test team to make flight abort decisions. The 2012 AR&C flights used a downward facing optical camera (Illunis RMV-4201 with an Active Silicon Phoenix D48CL frame grabber). For the 2013 HAZ flights, the Illunis was removed and a stereo camera facing downward and partially in the direction of flight was installed. Multiple GoPro Hero 3 cameras with 1080p video resolution are on the lander, including one looking straight down.

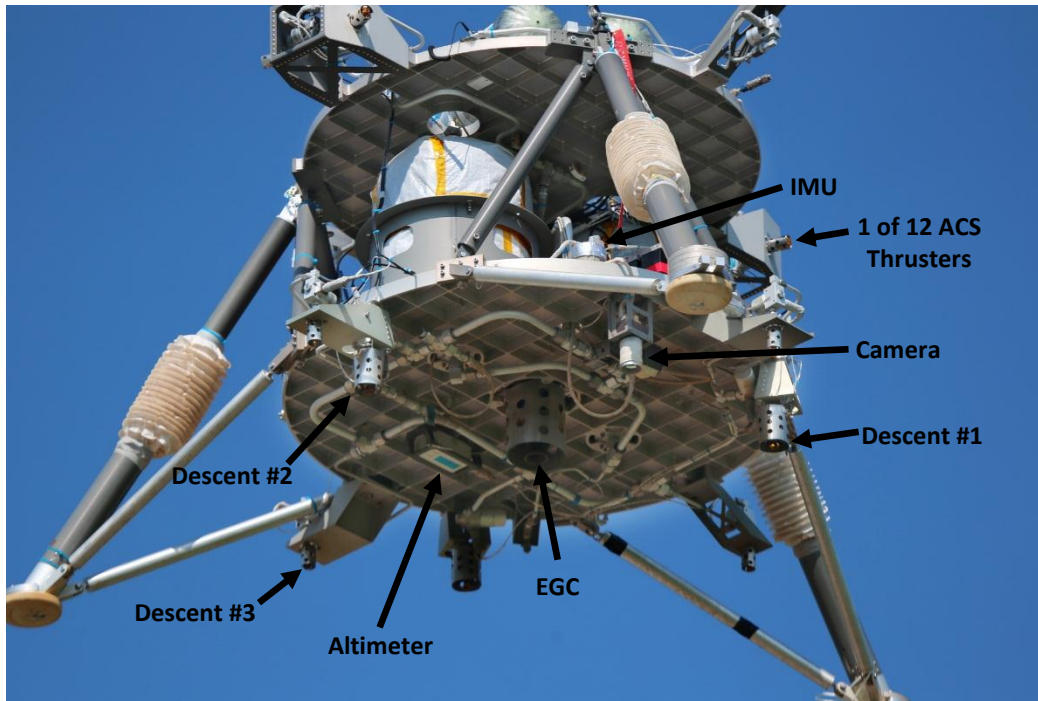


Figure 2. Locations of Mighty Eagle Sensors & Thrusters.

The Lander Avionics Unit (LAU), by Southwest Research Institute, uses a BAE radiation hardened RAD750 processor with 256 Mbytes of memory running at 132 MHz. This was chosen because of its similarities to existing space-flight processors.

2.2. *GNC*

There are two, independent Guidance, Navigation and Control algorithms run by flight software. GNCA, developed by APL (McGee et al. 2011a) is the active algorithm during all nominal conditions. GNCB, developed by SAIC (now Leidos), is an independent algorithm that takes control from GNCA when a “Land Now” abort condition is triggered on-board, or commanded via the test team. When this occurs, GNCB directs a vertical descent, attempting to cancel any lateral translation that may exist. Also, prior to liftoff, GNCB estimates the initial attitude, earth rotation and vertical accelerometer bias and passes this information to GNCA.

For each flight, a time-tagged sequence of commands is uplinked to the flight software. These commands can be used in differing combinations to produce the various flight modes (hover, lateral translation, diagonal ascent/descent, pure vertical descent) (McGee et al. 2011b). For hazard avoidance a 45 meter translation across the terrain field was performed. The ability to ascend and descend while translating (diagonally) was required in order to successfully traverse the terrain field within the flight time capability of the Mighty Eagle.

The translation navigation solution uses the IMU and the radar altimeter data. Data from the radar altimeter when the EGC is firing has been problematic. When the altitude is below 4 m, virtually all of the measurements are “good”. Above 20 meters between 0% and 33% of the measurements are “good”. To accommodate the large number of “bad” measurements a “gate” was added to monitor the altimeter signal. Altimeter measurements are rejected if they are more than a preset distance from the previous navigation state. Because of IMU accelerometer drift, fewer altimeter measurements meet the gate criterion towards the end of a flight. It is believed that modifying the radar frequency and/or adjusting the radar filters could

significantly improve altimeter performance. However, as the total system provided adequate performance, modifications to the altimeter were not pursued.

3. AR&C Testing

MSFC has a long history using Autonomous Rendezvous & Capture (AR&C) for docking/capture/berthing operations. Applications for AR&C on a free flying vehicle include satellite servicing, debris mitigation, sample return, and near earth object proximity operations. Using the existing Illunis camera an AR&C algorithm was added to flight software in 2012. By implementing the AR&C algorithm as an outer loop guidance module, we were able to leave the GNCA code unchanged. By not altering GNCA, the simulation & flight validation work did not need to be repeated. “Closed Loop AR&C” then means that the AR&C algorithm will deliver an external guidance command to GNCA whenever a valid AR&C solution is reached. A “valid” solution is one where the AR&C algorithm finds the target and one that is within pre-defined acceptable limits of the range. If no valid AR&C solution is found the guidance command uses preloaded coordinates.

The AR&C target consisted of four circles painted on steel plates (Fig. 3). With knowledge of the diameters and the relative positions of the target circles, the image-processing algorithm solves for the target in camera coordinates. The code then uses the inertial position estimate from GNCA and transforms the target from camera to inertial coordinates. This guidance solution is then passed to GNCA as its commanded position.



Figure 3. Mighty Eagle above the AR&C target in the West Test Area of MSFC. Photo extracted from a video camera mounted on a quad-copter.

3.1. AR&C Software

The RAD750 flight processor has limited processing power and the addition of AR&C processing created problems. Due to the computational limitations of the RAD750, a significant amount of effort was expended optimizing execution speed of the AR&C code to enable real time application. The image processing pipeline originally included support for multiple features, which made it more robust to image distortions and noise. However the increased functionality significantly taxed the flight processor. By controlling the field-of-view, which permitted removing extra processing, it was practical to achieve multiple solutions in flight (Fig. 4).

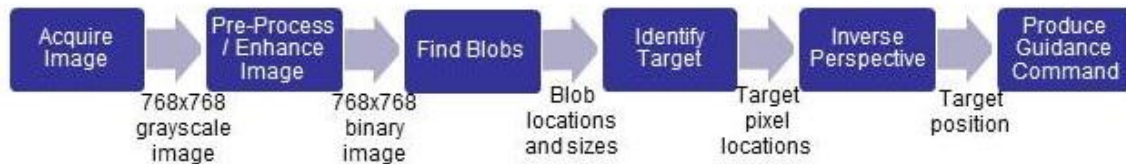


Figure 4. Summary of the AR&C algorithm flow.

The AR&C cFE (Core Flight Executive) application was designed to utilize the simulation and autocode build process already established with GNCB and the cFE (McGee et al. 2013). Thus, the AR&C application functionally segmented the image processing and command generation from the hardware.

By integrating the AR&C algorithms into a MATLAB/Simulink block, we placed the model into the validated Mighty Eagle simulation and could analyze the AR&C algorithm performance. First, the AR&C algorithm was refined and tuned using the desktop simulation. It was then autocoded and tested on the HWIL simulation. The HWIL simulation uses the flight processor and hardware, but the inputs (IMU, altimeter, etc) and outputs (thruster commands, EGC commands, etc.) are fed to a real-time instance of the Mighty Eagle dynamics model. Camera images, synthetic and real, were supplied to the AR&C application as the simulation was executed. Results were evaluated using the replay/post-test analysis tools.

3.2. Lessons Learned

After tethered tests ensured that modifications to flight software did not impair normal flight control, the AR&C system was tested on four Mighty Eagle flights in August and September of 2012. The first AR&C flight test was “open loop” at an altitude of 10 meters with 10 inch diameter circular white targets. “Open loop” means the AR&C algorithm was running, but the guidance command was not being passed to GNCA. No AR&C solutions were generated on this first flight. Post flight analysis of the images showed very few pixels where “white enough” to pass the image threshold of 240, in a 0 – 255 dynamic range. The threshold was changed to 100 and the minimum blob area was changed from 50 to 200 pixels

With these changes, the second AR&C flight was “closed loop”. On this flight, nine AR&C solutions were generated and the vehicle autonomously flew to the in-flight generated solutions. The entire target was only in view for approximately 16 seconds during which the system averaged approximately one solution every 1.8 seconds. The final two AR&C flight tests were done at an altitude of 30 meters with 24 inch diameter targets. AR&C flight 3 was open loop and 10 valid solutions were generated. The profile for the last AR&C flight was identical to flight 3 the AR&C was in closed loop mode. Eleven valid AR&C solutions were generated and the Mighty Eagle landed 60 cm from the target. This concluded a successful demonstration of AR&C in a robotic lander test-bed.

4. Hazard Avoidance (HAZ)

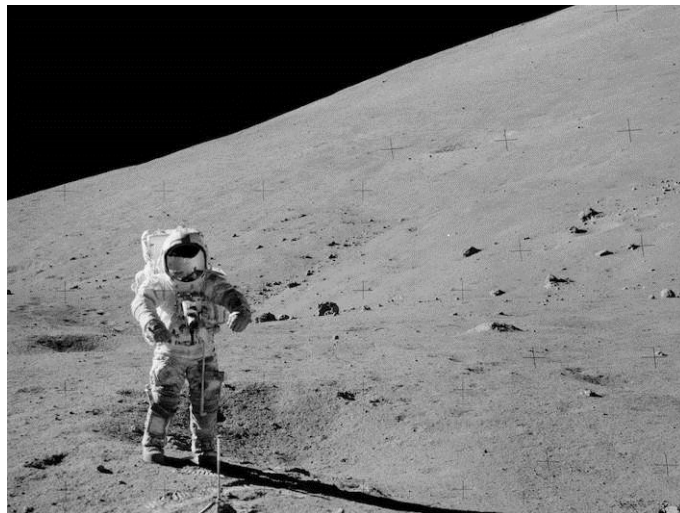
In 2013 the AR&C effort was extended to develop hazard avoidance. An onboard, commercial, stereo camera (BumbleBee 3 from Point Grey) was to identify hazards (boulders, slopes, craters) in a lunar terrain field. The BumbleBee 3 uses three cameras (Sony ICX445) mounted into a frame. These cameras are gray-scale, sensitive to the visible portions of the EM spectrum, and 1280 x 960 detector arrays. The baseline distance between the outer cameras was 24 cm. Spatial resolution at nadir was 0.61549 mrad.

The initial plan was to autonomously identify a safe landing zone in a lunar-like test field using onboard sensors while in flight, then have the hazard avoidance algorithm pass an external guidance input to GNCA. This is similar to how AR&C work was implemented. However, landing in a terrain field would probably do significant damage to the lander because the exhaust from the thrusters would excavate the terrain field (Metzger et al. 2011). Steel plates in the terrain field would seriously compromise the terrain field fidelity, raising doubt about the utility of test results. Instead, launch and landing zones made from welded steel plates were constructed on opposite sides of the terrain field. Flight profiles varied but all hazard avoidance flights were commanded to land on the steel landing zone, 45 meters downrange. Initial alignment of the Mighty Eagle was vital for these long downrange translations in order to set down on the steel plated landing area.

4.1. Terrain Field

In order to test the hazard avoidance system an outdoor terrain field was needed. Major design criteria were to fill the field of view of the Bumblebee at 30 meters altitude, simulate the lunar surface, and be as inexpensive as practical. The design was to be adequate for, but not limited to, the hazard avoidance testing.

To a first order approximation, the lunar surface can be described as having an average particle size of approximately 50 μm (Carrier 2003), and composed of glass and the minerals plagioclase, pyroxene and olivine (Papike 1991). For practical purposes there is very little variation in color or reflectivity. The fine particle size means surface variation, perceived by a human as “texture”, is effectively not present or is greatly subdued from terrestrial norms. Imposed on this featureless surface are occasional rocks, boulders, and topography caused by hypervelocity impacts. Changes in topography are detectable due to changes in illumination angles. However, many crater edges are very rounded. As a result, when the Sun is near zenith smaller craters become very problematic to detect visually (Fig. 5). In many locations the only spatially sharp features are the boundaries of shadows from rocks.



The mission profile called for flight altitudes of 20 – 30 meters, providing a pixel size of 12 – 18 mm respectively. To prevent detection of heterogeneity caused by having more or less shadow in a pixel, i.e. to prevent detection of a surface texture, particle size for the terrain field needed to be approximately 6 mm or finer. As a lower limit to particle size, abundant particles below 10 μm are a respiratory health risk and are difficult to work with. The health risk is especially significant if the dust contains quartz or other hazardous

silica phases. It was therefore decided to purchase simulant with particle sizes small enough to be representative of the lunar regolith yet large enough that a hazardous site was not created.

After extensive review of possible sources and transportation options (Rickman 2013) material from the Merriam Crater, north east of Flagstaff, Arizona was purchased from Miller Mining of Flagstaff. This is the source of material used to make the JSC-1 series of lunar simulants. 200 tons of “Black Cinder Sand” and 25 tons of “Biosphere Sand” were shipped by end- and belly-dump trailers. The vendor supplied particle distribution for the products is given in Table 1.

Table 1. Particle size distribution for simulant material used in the Lunar Surface Terrain field and typical coarse lunar values are the lower one standard deviation curve from Carrier (2003).

Inches & Mesh	Millimeters	% Passing		Typical Coarse Lunar
		Black Cinder Sand	Cinder Fines “Biosphere”	
3/8”	9.5	100		
1/4”	6.3	99		100
4	4.75	92		93
8	2.36	57	100	91
10	2.00	48	99	89
16	1.18	32	87	84
30	0.60	22	71	78
40	0.425	18	65	74
50	0.30	15	56	70
100	0.15	9	35	68
200	0.075	5.5	19.6	64

The site of the terrain field was treated with herbicide, covered with geofabric and covered by the coarser sand to an average depth of approximately 12 cm. The finer sand was used to top the surface (Fig. 6).



Figure 6. The MSFC Lunar Surface Terrain field. There are 9 craters and 11 boulders. A: steel plates for launch. B: steel plates for landing, with lander. C: folded infield tarp used to cover terrain field when not in use.

Boulders are a hazard for the lander, should it be necessary to abort and land on the terrain field. Therefore the boulders needed to be either light enough to be blown out of the way by the descending lander, or be easily crushed by the lander's weight. It was decided that commercially available, fake rocks, made from polyethylene with admixed crushed stone, were suitable. While hollow and lightweight, these have appropriate external morphologies and aspect ratios to simulate lunar boulders (Demidov and Basilevsky 2013). To make them appear spectrally appropriate the boulders were liberally coated with Weldwood brand contact cement, and while wet heavily dusted with fines sieved out of the Biosphere sand. As the contact cement dries it pulls the particles together, greatly reducing any visibility of the cement. After being rinsed to remove very fine, loose dust, the boulders are easy to handle and effectively invisible on the terrain field except for their shadows. Eight of the boulders were coated; the three uncoated boulders are clearly visible in Fig. 6.

Craters were simulated by hoeing outward in a radial pattern. This permits creation of craters approximately 30 cm deep. To minimize risk to the Mighty Eagle in an emergency landing, they were placed away from the line of expected travel.

4.2. HAZ Software

The computational demands of hazard avoidance far exceeded the resources available on the RAD750; therefore, a laptop (Panasonic Toughbook 31) was mounted on the vehicle to serve as an image processor. This enabled us to readily implement computationally intensive algorithms, run as a cFE application, on the Mighty Eagle. Also, the team wanted to demonstrate multiple cFE instances, with different operating systems, on a single vehicle. To bridge instantiations of cFE, Goddard Space Flight Center developed the

Software Bus Network (SBN) cFE application. SBN was prototype software and had not been used on a flight project.

The general software architecture has two instances of cFE running, one in VxWorks and the other in Linux (Fig. 7). With this architecture, the previously validated flight computer software did not need to be modified to support this new test series. Integrating the laptop on the vehicle required only an ethernet switch and a mounting bracket.

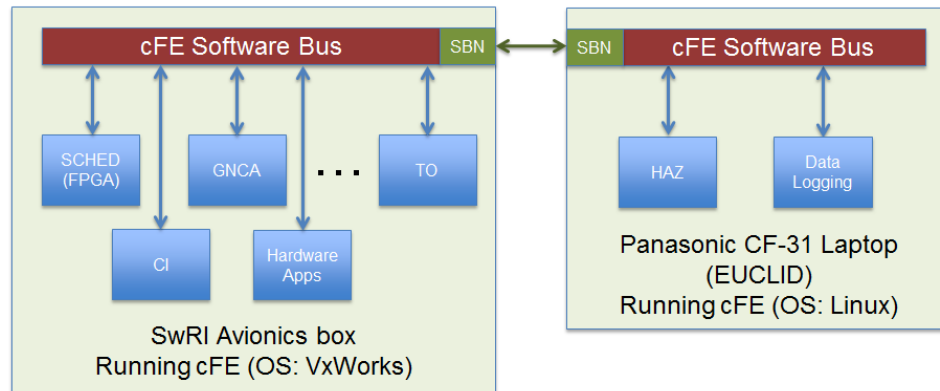


Figure 7. Hazard avoidance cFE architecture

The HAZ cFE application was designed in the same manner as the previous AR&C tests, except the image processing algorithm was written in C++. Point Grey provided a SDK to communicate and capture images from the camera and one that could perform the stereo image processing. The hazard avoidance library was designed to expose a C-wrapper API; thus allowing the library to be called from multiple environments and build types. The library was included in the cFE flight software application, stand-alone test applications, a Simulink S-function, and could run on Linux and Windows. This allowed the same software library to be tested using the Mighty Eagle simulation, flight hardware, the replay capability, and custom test harnesses.

4.3. *Lessons Learned*

It is apparent that expansion capability built into the avionics suite of a developmental lander is important. Some desired testing could not be performed with the test article due to the lack of available I/O ports. Future testbeds similar to the Mighty Eagle should incorporate I/O expansion capability. For the hazard avoidance test series, having an accessible ethernet port would have saved engineering design time and payload weight.

The depth resolution of the BumbleBee camera at the flight altitude was not adequate for the intended purpose. In part, this was due to the flight parameters, but it was also due to the nature of the lunar surface as reproduced by the terrain field. The logic used by the Point Grey SDK works using image disparity. For a surface such as the Moon's, there are too few features for the vendor-supplied algorithms to work with.

More importantly, the fine particles of the lunar terrain field were dispersed upward by the pulsed descent thrusters and the constant EGC thrust far too quickly for practical feature detection to occur (Fig. 8 and 9). While surface interaction of exhaust with the lunar surface is well known and studied (Metzger et al. 2011), the opacity, temporal frequency, and magnitude of “dust waves” were not predicted. How these features would translate to a different atmosphere or vacuum with different thrusters, flight profiles and regolith is unknown. Therefore, even if problems in image stabilization, vertical resolution and data processing rates can be solved, the formation of random turbulence features needs to be considered. This suggests that for close approach monitoring, use of an electromagnetic wavelength substantially longer than the average particle size, might be preferable.



Figure 8. Large amount of dust generated by the plume from the Mighty Eagle on hazard avoidance as it traverses the terrain field at an altitude of 20 meters. Photo extracted from a video camera mounted on a quad-copter.

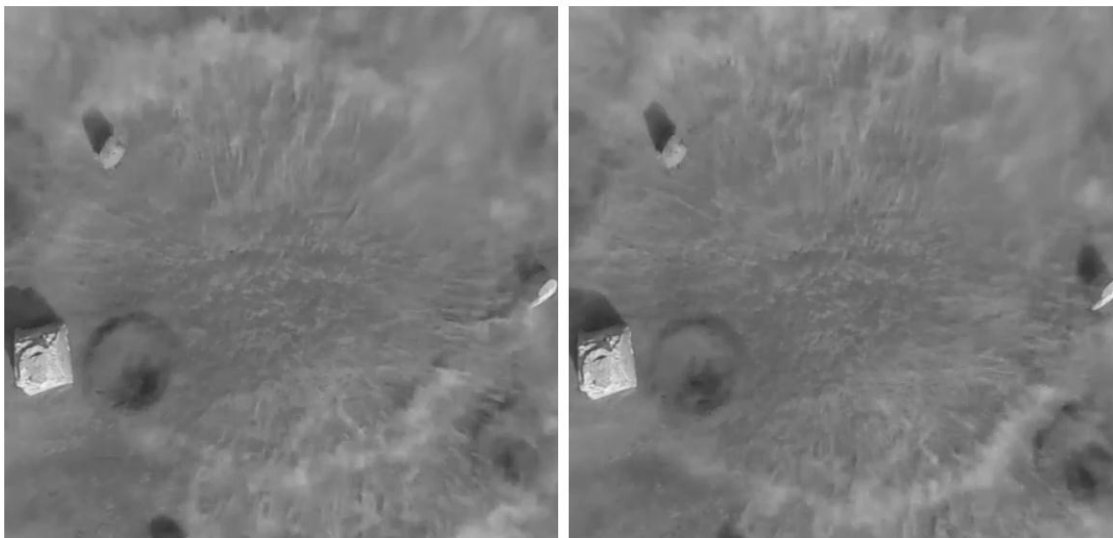


Figure 9. Adjacent video frames, approximately 1/30th of second separation, from the down looking video camera on the lander. Note substantial movement and change in the dust waves.

5. Moon Express Testing

The Mighty Eagle team partnered with Moon Express, Inc. (MEI) under a Space Act agreement to test guidance software by MEI. Their software was a new development and the Mighty Eagle was its first application on a physical vehicle.

The Moon Express Guidance (MEG) software package was implemented on the Core Flight Executive (cFE) architecture, which simplifies integration into the overall software architecture. In the software configuration used in the 2011 test flight series, the APL-developed GNCA software performed all of the vehicle's guidance, navigation, and control tasks. The 2012 and 2013 test flight series featured additional cFE modules, which performed, respectively, the optical automated rendezvous and hazard avoidance tasks. These modules worked by providing a command to GNCA's guidance system which overrides the predetermined course of flight and specifies an alternate "go-to" position, acting as an outer-loop controller over the existing guidance system.

In contrast, the MEG software (Fig. 10) had to completely replace the standard GNCA guidance package, setting the vehicle's trajectory by taking over both pointing guidance and descent motor pulsing. Pointing guidance was overridden by placing GNCA into an attitude tracking mode, where attitude commands from MEG were fed into GNCA's control system. Then, thanks to GNCA's modular nature, the team was able to disable its vertical control and via a software interlock use descent motor commands provided by MEG which provided both vertical guidance and control. GNCA retained attitude control and navigation duties, which along with mass estimates provided by GNCB, were all passed to MEG to close the guidance loop. MEG software was modified to both accept and output the CCSDS packet formats used elsewhere on the vehicle for these messages.

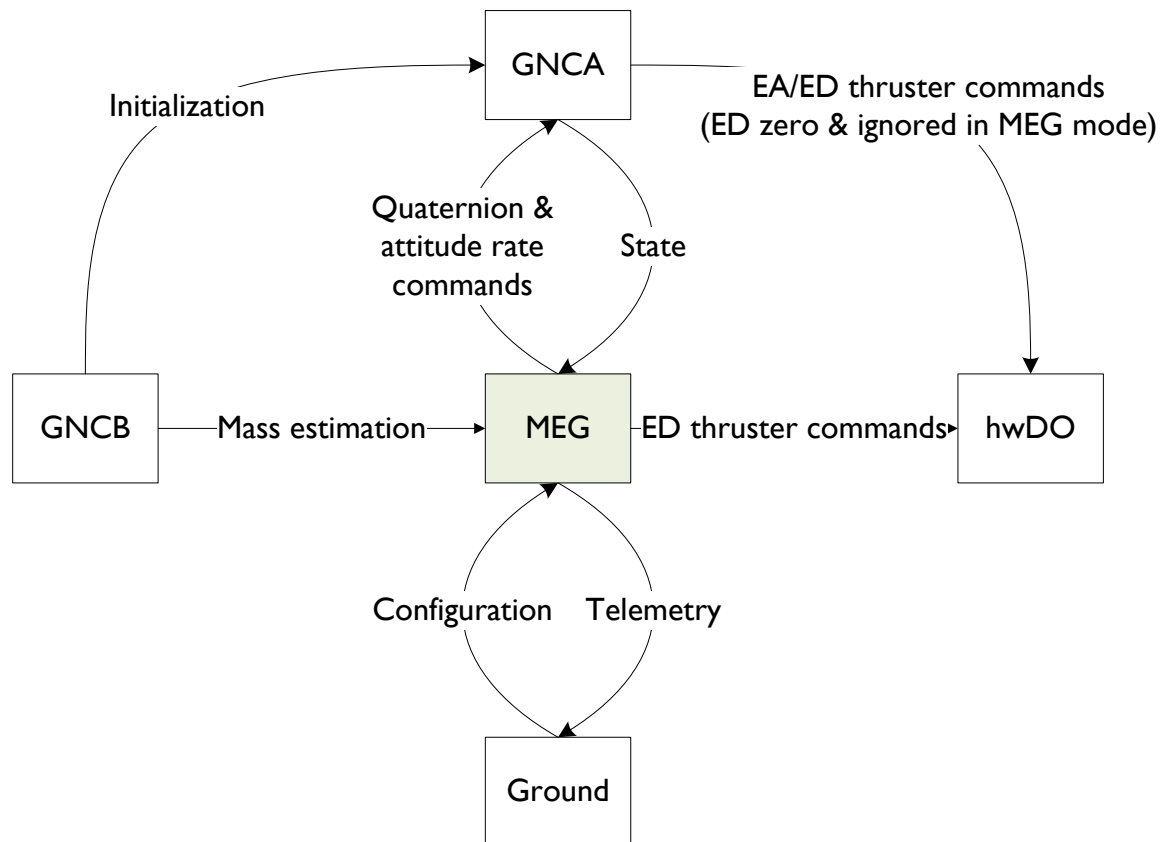


Figure 10: High-level architecture of the Moon Express Software for MEG flights.

Five commands were added to the Mighty Eagle time-tagged command script to support MEG operations. Three commands were passed to the MEG application to initialize MEG parameters, change modes, and to direct the MEG application into its flight profile table. The other two commands gave MEG control of the Mighty Eagle descent thrusters and enabled the MEG-computed attitude command to be passed to GNCA.

Testing of the MEG software was an iterative process. The Mighty Eagle flight profile is controlled via a time-tagged command script with position and velocity control commands issued on one-second boundaries. In contrast, the MEG software team developed a table that was compiled into the flight software to achieve the desired flight profile. The table contained position (x,y,z) and velocity (xdot,ydot,zdot) targets as well gains for each coordinate. Once the table was compiled into the Mighty Eagle flight software build, the avionics and dynamics simulator were rebooted and the flight profile was then executed in the simulation environment. Screen shots were taken of the resulting GNCA position and velocity plots and emailed to the MEG team along with the recorded GNCA/MEG high rate data from the simulated flight. Following a quick assessment, the MEG team emailed changes to gain settings (various combinations of vertical & lateral, velocity & position) in the table, which were incorporated, re-compiled, and tested again in the simulation environment. Once a table produced a simulation flight that matched the desired flight profile (position and velocity drifts within GNCB defined limits), the simulation was executed two more times to ensure reproducibility.

MEG software was active on three Mighty Eagle flights. The software first flew in open loop mode (MEG attitude guidance was not sent to GNCA and control of the descent thrusters was not enabled) on the HAZ03 flight. Following completion of Mighty Eagle AR&C testing, the project approved a tethered flight with MEG operating in closed loop mode. The flight profile used was the same profile that the Mighty Eagle had flown routinely under GNCA control: 0.7 meter altitude, 10 second hover, and 14 second total flight time. After a review of data from the MEG01 flight, it was decided to add an untethered flight with MEG in closed loop mode. For this flight, a profile similar to an earlier Mighty Eagle flight was designed: 3 meter altitude, 12 second hover, 21 second total flight time.

5.1. Lessons Learned

The single most difficult interface issue encountered while integrating the MEG software was effectively communicating the vehicle and navigation reference frames. Relatively late in the development cycle, the Mighty Eagle and Moon Express teams identified that while we believed we had agreed on the reference frames, we were not in fact working to the same set of frames, which in turn led to substantial problems with software simulations up to that time. In our communications, vehicle reference frames were specified either as text (by the Mighty Eagle team) or as vectors drawn on an orthographic view of the vehicle (by the Moon Express team). In the future, the Mighty Eagle team should provide to third parties reference images with clearly marked axes, origin, and reference features.

Similarly, there was confusion about the navigation frame. GNCA uses a simplified, but idiosyncratic frame. This frame has no fixed orientation, be it earth-centered, earth-fixed (ECEF), local Cartesian north-east-down, or analogous frames. Rather, it varies based on the initial orientation of the vehicle during a flight's navigation initialization. Future projects should seek to avoid this frame as a public interface due to its confusing nature.

6. NanoLaunch IMU

NanoLaunch is a MSFC led project, which is developing a small (1 to 10 kg payloads), low-cost launch vehicle. The fact that the Mighty Eagle was conducting flight tests provided the NanoLaunch GNC team an opportunity to test candidate sensors on these flights. On the two Moon Express flights a "payload" containing two candidate NanoLaunch sensor suites was fastened to the lower deck of the Mighty Eagle. These suites, each with an IMU and GPS, were integrated into a single package. This payload was completely independent of all Mighty Eagle avionics: containing its own battery and data recording device. Two hardware approaches to avionics design were included: a Cortex M3 (Arduino Due) interfaced with an Epson M30-5 IMU and a Skytraq Venus638FLPx GPS receiver, and a Cortex M9 (BeagleBone Black)

interfaced to a VectorNav VN-100 IMU and GlobalTop FGPMMA6H (MTK339 chipset) GPS module. The primary objective of these tests was to examine the performance these relatively low cost sensor suites on actual rocket powered flights. The performance of the sensors will serve as valuable inputs to the NanoLaunch team.

7. Conclusions

Flight demonstrations of Autonomous Rendezvous & Capture in 2012 and Hazard Avoidance in 2013 were conducted at the Marshall Space Flight Center using the Mighty Eagle robotic lander. These flights provided practical experience in specific test objectives and with GNC during terminal approach. The AR&C task built on a substantial existing expertise in this area and applied it for the first time to a robotic lander. Demonstrating hazard avoidance task was something new for MSFC. While the stereo vision system did not successfully demonstrate hazard avoidance in flight we did show via post-flight analysis of the stereo images that with software tuning it could be done. Several changes to both software and hardware were identified that will make this a more robust system. A high-fidelity lunar terrain field was built as part of this effort and promises to serve as a valuable asset in a wide variety of future test programs. The final two flight tests in 2013 were dedicated to the validation of GNC algorithms for Moon Express.

8. Acknowledgements

The Mighty Eagle flight test programs in 2012 and 2013 would not have been possible without the contributions of many talented individuals. Cindy Stemple in the Flight Projects office provided valuable support and encouragement. Eric Ordonez and Sorita Wherry provided system engineering leadership. Our chief engineers, Rene Ortega and Greg Frady, provided technical leadership to the team. Bryan Robertson, Tom Bryan and Ricky Howard provided wide ranging avionics support. Tim McGee from APL provided GNC insight. Jake Parton was our lead text engineer. The red crew which prepared the Mighty Eagle for flight included Kevin Thompson, Danny Duke, Craig Havens, Daniel Briscoe, Chris Morrow, Blake Thompson, Jarrod Lemaster, Ben Pattillo, Daniel Clanton and Lance Walker. Adam Burt did all of the mechanical design work for the Hazard Avoidance System. Scott Schutzenhofer and Robert Rutherford managed the challenging task of getting 225 tons of lunar simulant from a quarry in Arizona to Huntsville. Patrick McVay and Richard Johnson led the facilities team that built the lunar terrain field. Hunter Williams was our propulsion expert. Kyle Daniel kept us all safe. Dan Lazor and Dane Childers wrote the AR&C algorithms. Scott Akridge, Carl Benson and Jim Cecil provided software support. Todd Freestone and Mathew McGrath handled all communications and ESD issues. Garrick Merrill, Chris Becker, Adam Kimberlin and Peter Ma flew and acquired video from the quadcopter. Evan Anzalone coordinated the NanoLaunch IMU evaluation effort.

9. References

1. Carrier, W. David III. 2003. "Particle Size Distribution of Lunar Soil." *Journal of Geotechnical and Geoenvironmental Engineering* 129 (10): 956. doi:10.1061/(ASCE)1090-0241(2003)129:10(956).
2. Demidov, N E, and A T Basilevsky. 2013. "Rock Fragments Height/Diameter Ratio as Measured on the Lunokhod and Apollo Surface Panoramas." In *44th Lunar and Planetary Science Conference*, Abstract #1859. Houston: Lunar and Planetary Institute.
3. McGee, Timothy. Cole, D. Artis, B. Morse, C. Reid, G. Chavers, C. Stemple, J. Bassler, L. Hill, B. Mulac. 2011a. "NASA's Robotic Lunar Lander Development Project: Initial Flight Testing Results of a Robotic Lunar Lander Test-Bed", IAC-11.A3.2B.5 Proc. of the 62nd International Astronautical Congress, Cape Town South Africa, October 2011.
4. McGee, Timothy, J. Kaidy, D. Reid, G. Oxtan & M. Hannan. 2011b. "Guidance, Navigation and Control Development for a Robotic Lander Testbed", Proc. of the 33rd Annual AAS Guidance and Control Conference, Breckenridge, CO, February 2011.
5. McGee, T., J. Artis, D. T. Cole, D. Eng, C. Reed, M. Hannan, D. G. Chavers, L. Kennedy, J. Moore & C. Stemple. 2012. "Mighty Eagle: The Development and Flight Testing of an Autonomous Robotic Lander Test Bed", Johns Hopkins APL Technical Digest, Volume 32, Number 3, December 2013.

6. Metzger, Philip T., Jacob Smith, and John E. Lane. 2011. "Phenomenology of Soil Erosion Due to Rocket Exhaust on the Moon and the Mauna Kea Lunar Test Site." *Journal of Geophysical Research* 116 (E6) (June 30): 1–22. doi:10.1029/2010JE003745.
7. Papike, J. J., Lawrence A. Taylor, and S. B. Simon. 1991. "Lunar Minerals." In *Lunar Sourcebook: A User's Guide to the Moon*, edited by Grant Heiken, David Vaniman, and Bevan. M. French, 121–182. Cambridge University Press.
8. Rickman, Douglas. 2013. Sources and Transportation of Bulk Low-Cost Lunar Simulant Materials. NASA TM TM-2013-21